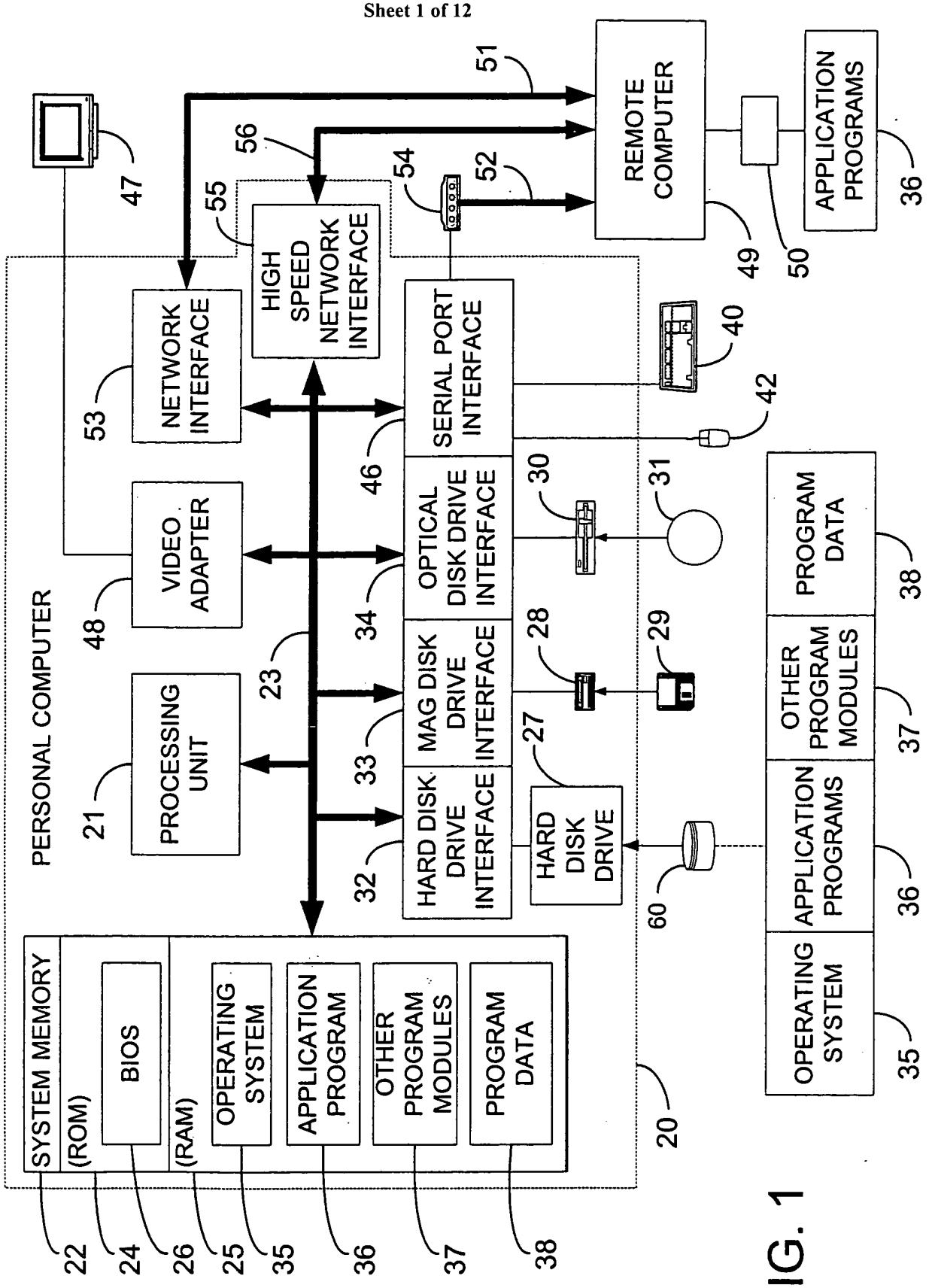
PERSONAL COMPUTER
SYSTEM MEMORY
(ROM)
BIOS
(RAM)
OPERATING SYSTEM
APPLICATION PROGRAM
OTHER PROGRAM MODULES
PROGRAM DATA
PROCESSING UNIT
VIDEO ADAPTER
NETWORK INTERFACE
HIGH SPEED NETWORK INTERFACE
SERIAL PORT INTERFACE
OPTICAL DISK DRIVE INTERFACE
MAG DISK DRIVE INTERFACE
HARD DISK DRIVE INTERFACE
HARD DISK DRIVE
REMOTE COMPUTER
APPLICATION PROGRAMS
PROGRAM DATA
OTHER PROGRAM MODULES
APPLICATION PROGRAMS
OPERATING SYSTEM
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
40
42
46
47
48
49
50
51
52
53
54
55
56
60

FIG. 1

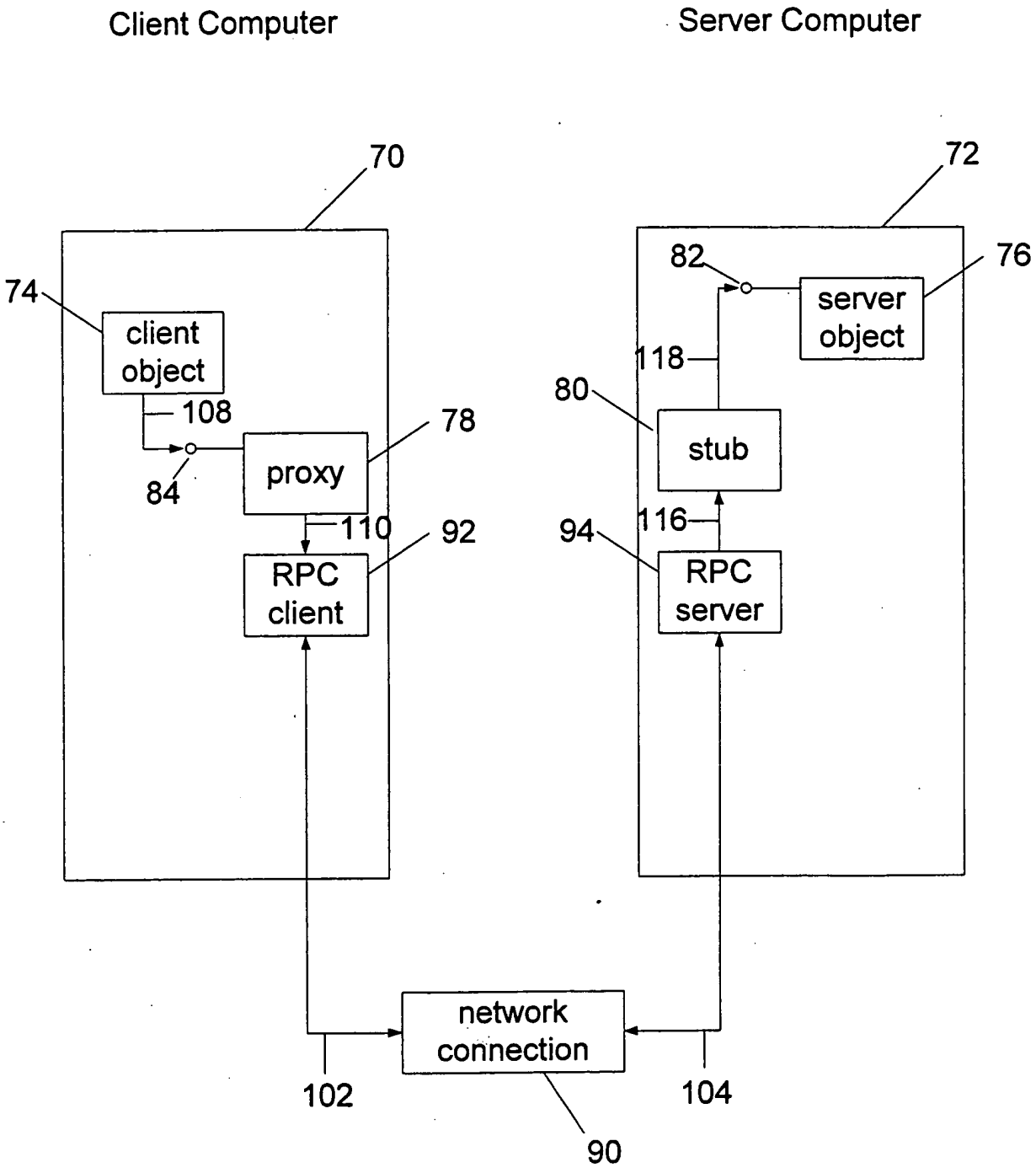
Client Computer
Server Computer
70
72
74
client object
108
78
proxy
84
110
92
RPC client
82
76
server object
118
80
stub
94
116
RPC server
network connection
102
104
90

FIG. 2

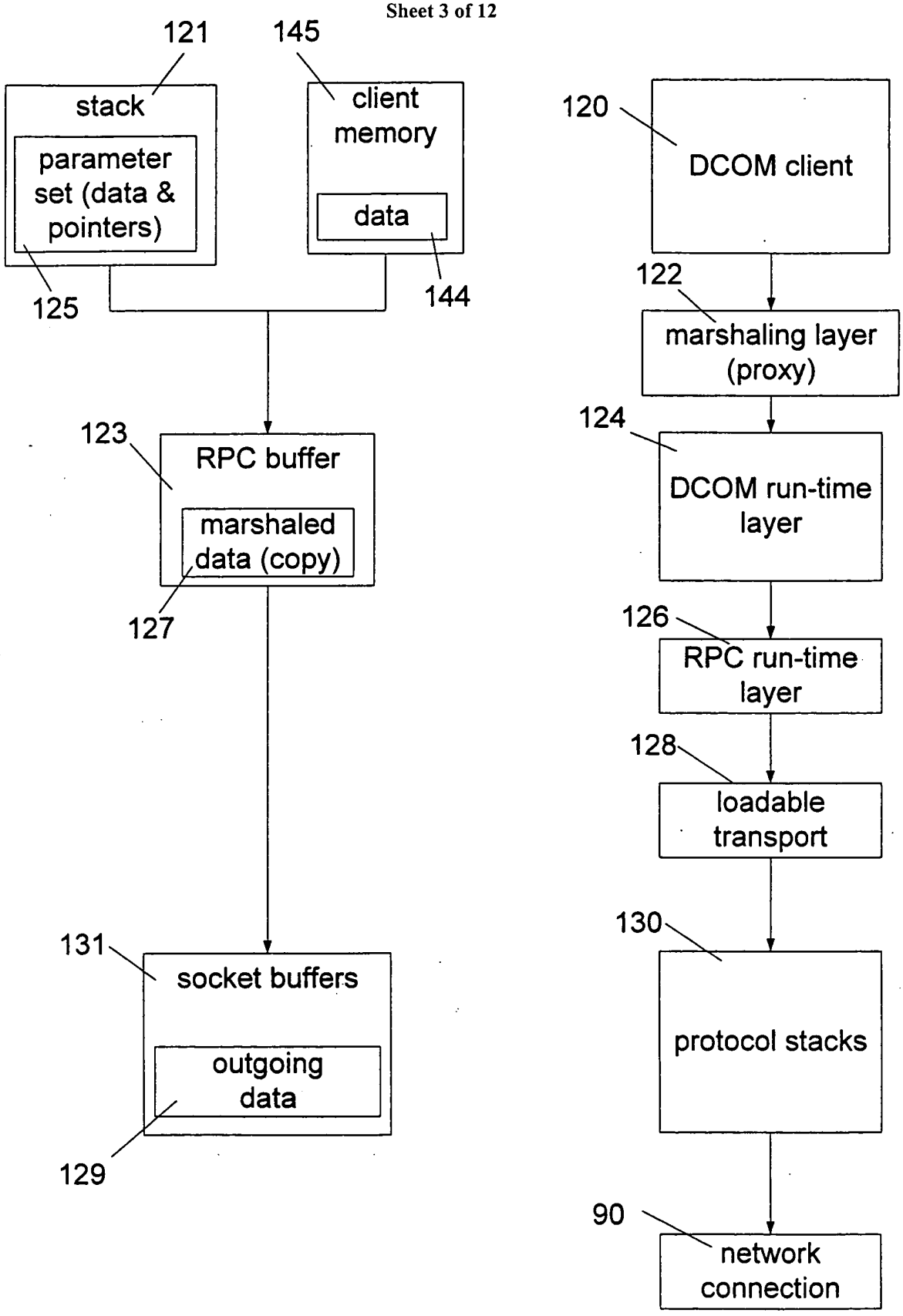
121
145
stack
parameter set (data & pointers)
client memory
data
125
144
120
DCOM client
122
marshaling layer (proxy)
124
DCOM run-time layer
123
RPC buffer
marshaled data (copy)
127
126
RPC run-time layer
128
loadable transport
130
protocol stacks
131
socket buffers
outgoing data
129
90
network connection

FIG. 3A

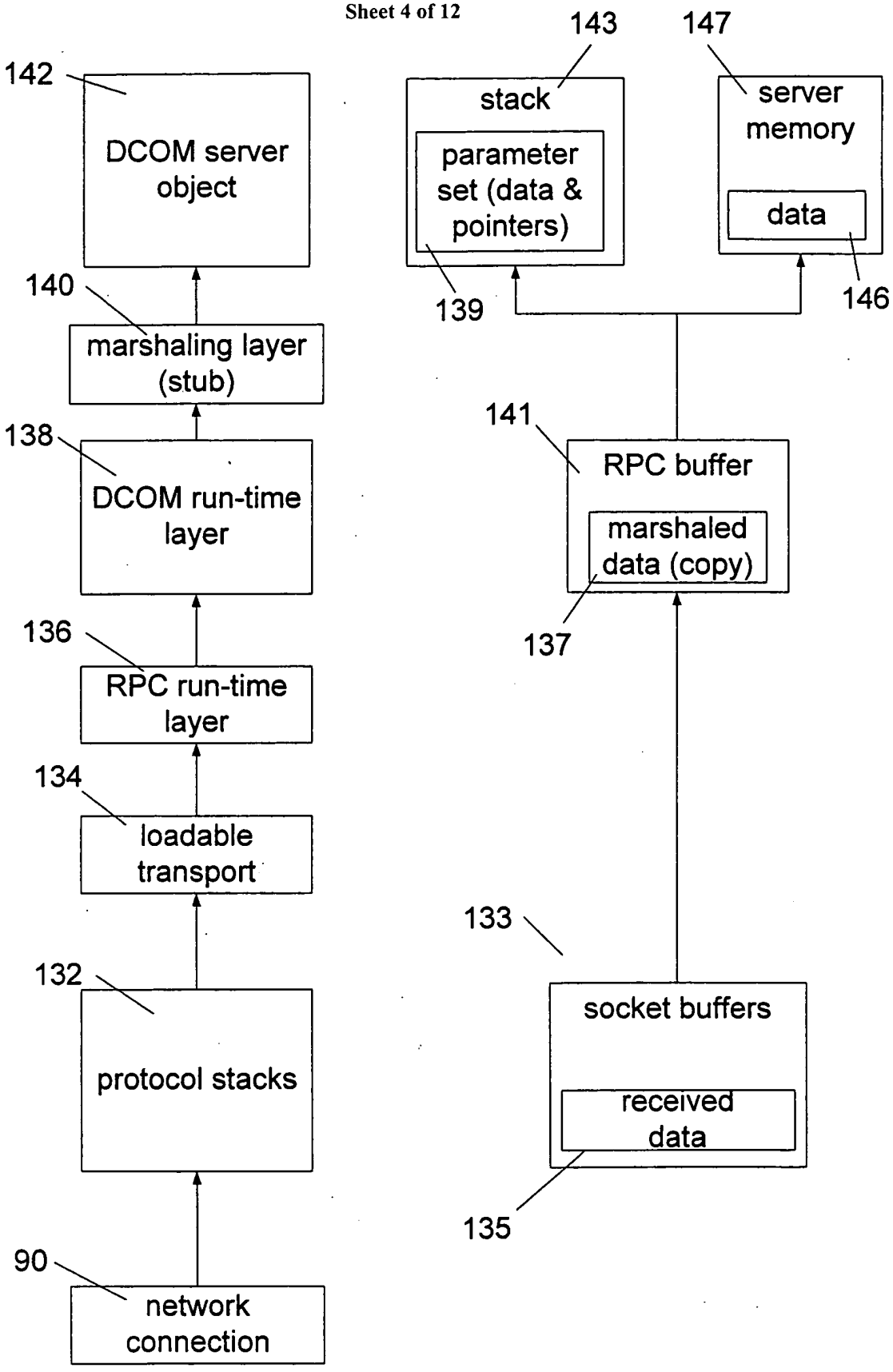
142
DCOM server object
140
marshaling layer (stub)
138
DCOM run-time layer
136
RPC run-time layer
134
loadable transport
132
protocol stacks
90
network connection
143
stack
parameter set (data & pointers)
139
147
server memory
data
146
141
RPC buffer
marshaled data (copy)
137
133
socket buffers
received data
135

FIG. 3B

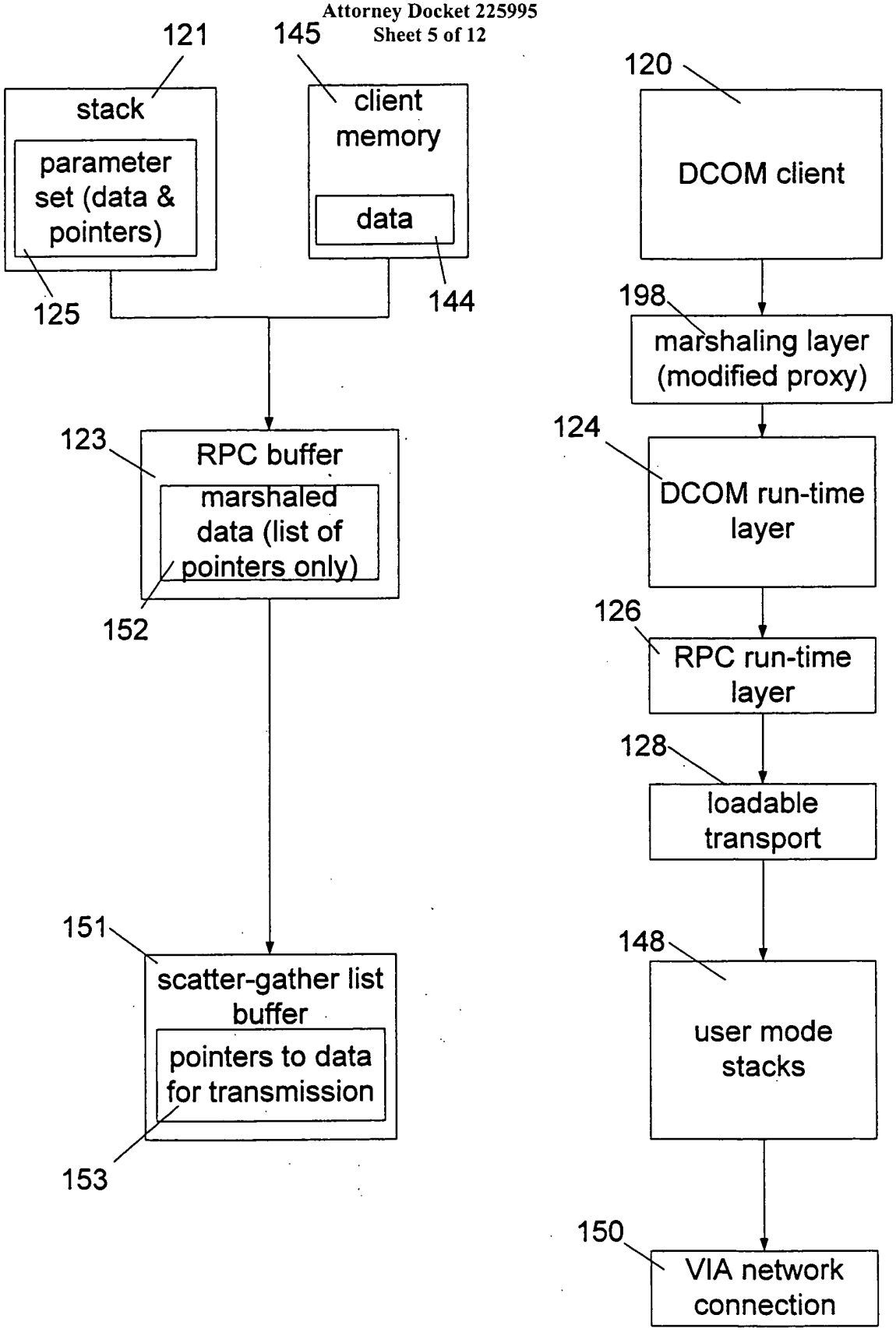
121
145
120
stack
parameter set (data & pointers)
client memory
data
DCOM client
125
144
198
marshaling layer (modified proxy)
123
RPC buffer
marshaled data (list of pointers only)
124
DCOM run-time layer
152
126
RPC run-time layer
128
loadable transport
151
scatter-gather list buffer
pointers to data for transmission
148
user mode stacks
153
150
VIA network connection

FIG. 4A

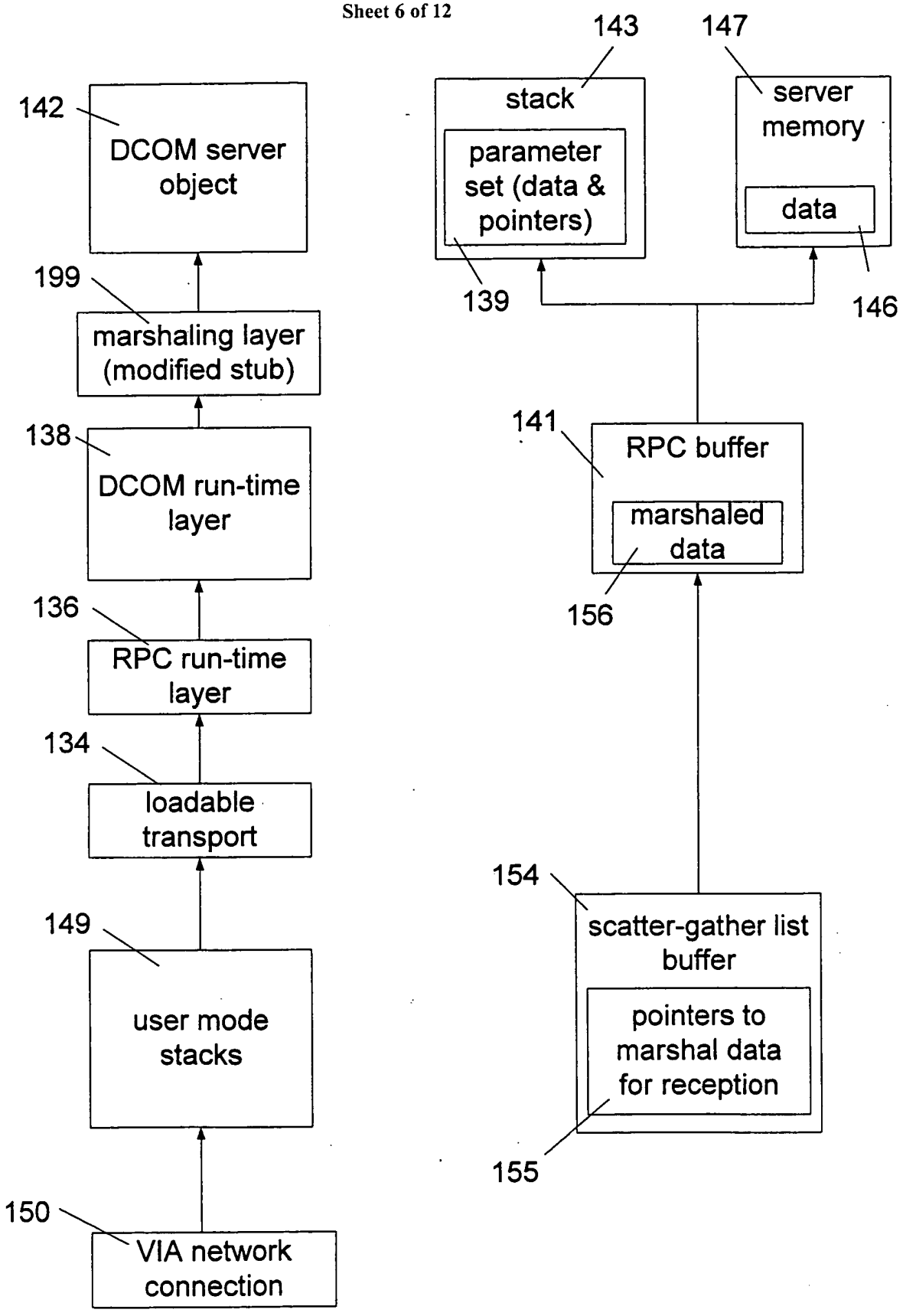
142
DCOM server object
199
marshaling layer (modified stub)
138
DCOM run-time layer
136
RPC run-time layer
134
loadable transport
149
user mode stacks
150
VIA network connection
143
stack
parameter set (data & pointers)
139
147
server memory
data
146
141
RPC buffer
marshaled data
156
154
scatter-gather list buffer
pointers to marshal data for reception
155

FIG. 4B

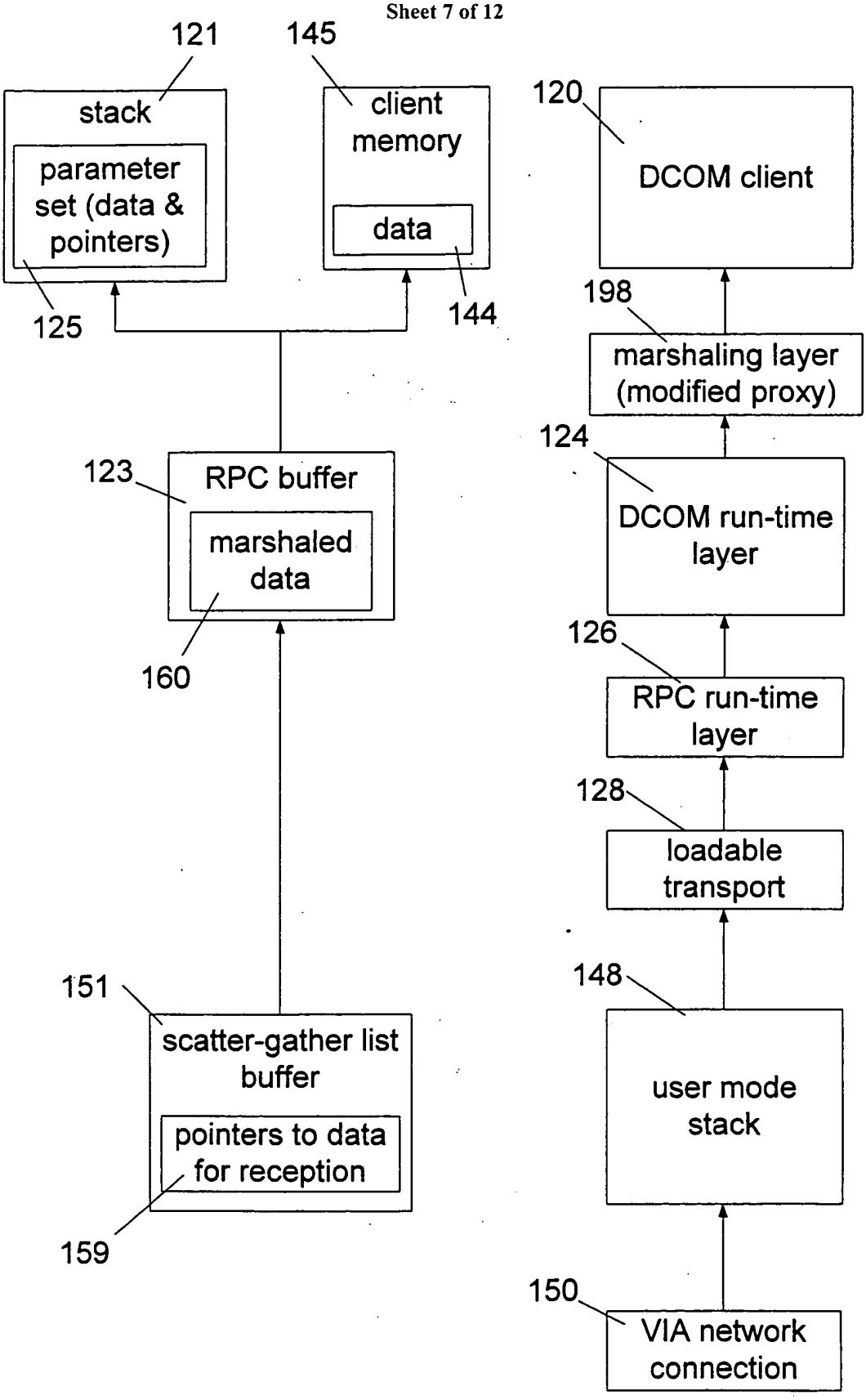
121
145
stack
parameter set (data & pointers)
client memory
data
125
144
120
DCOM client
198
marshaling layer (modified proxy)
124
DCOM run-time layer
123
RPC buffer
marshaled data
160
126
RPC run-time layer
128
loadable transport
151
scatter-gather list buffer
pointers to data for reception
159
148
user mode stack
150
VIA network connection

FIG. 5A

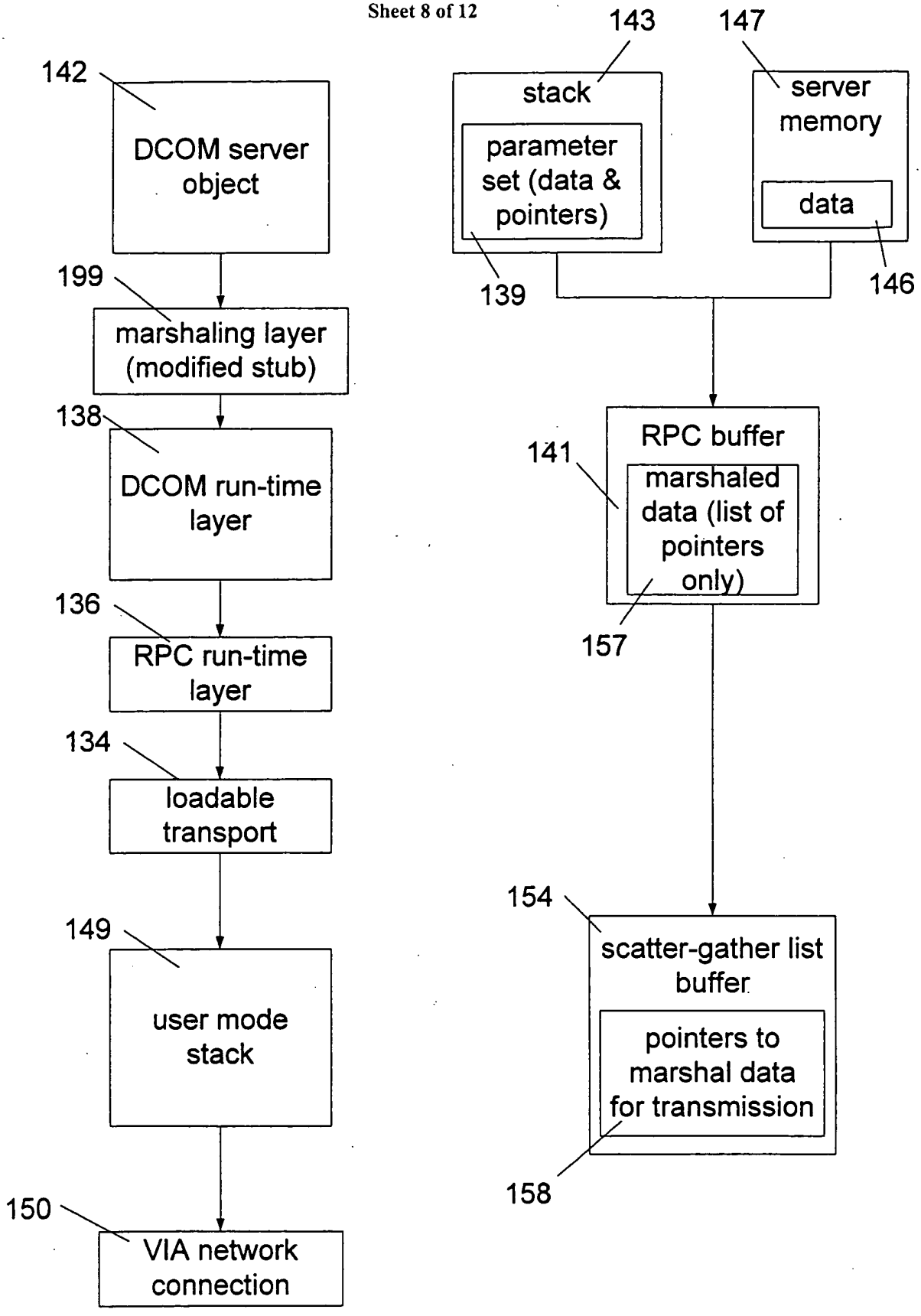
142
DCOM server object
199
marshaling layer (modified stub)
138
DCOM run-time layer
136
RPC run-time layer
134
loadable transport
149
user mode stack
150
VIA network connection
143
stack
parameter set (data & pointers)
139
147
server memory
data
146
RPC buffer
141
marshaled data (list of pointers only)
157
154
scatter-gather list buffer
pointers to marshal data for transmission
158

FIG. 5B

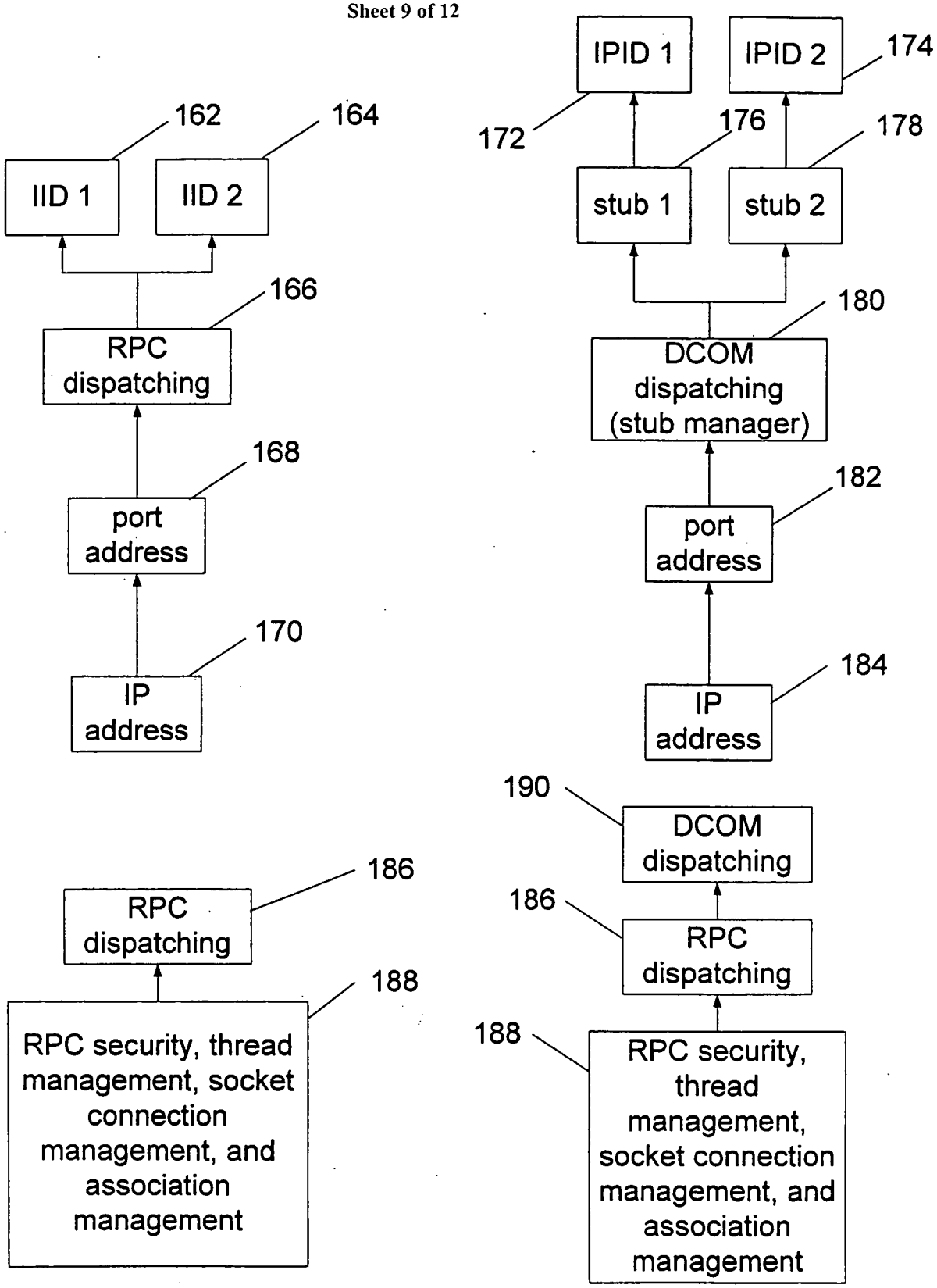
IID 1
IID 2
162
164
RPC dispatching
166
port address
168
IP address
170
RPC dispatching
186
RPC security, thread management, socket connection management, and association management
188
IPID 1
IPID 2
174
172
176
178
stub 1
stub 2
DCOM dispatching (stub manager)
180
port address
182
IP address
184
DCOM dispatching
190
RPC dispatching
186
RPC security, thread management, socket connection management, and association management
188

FIG. 6

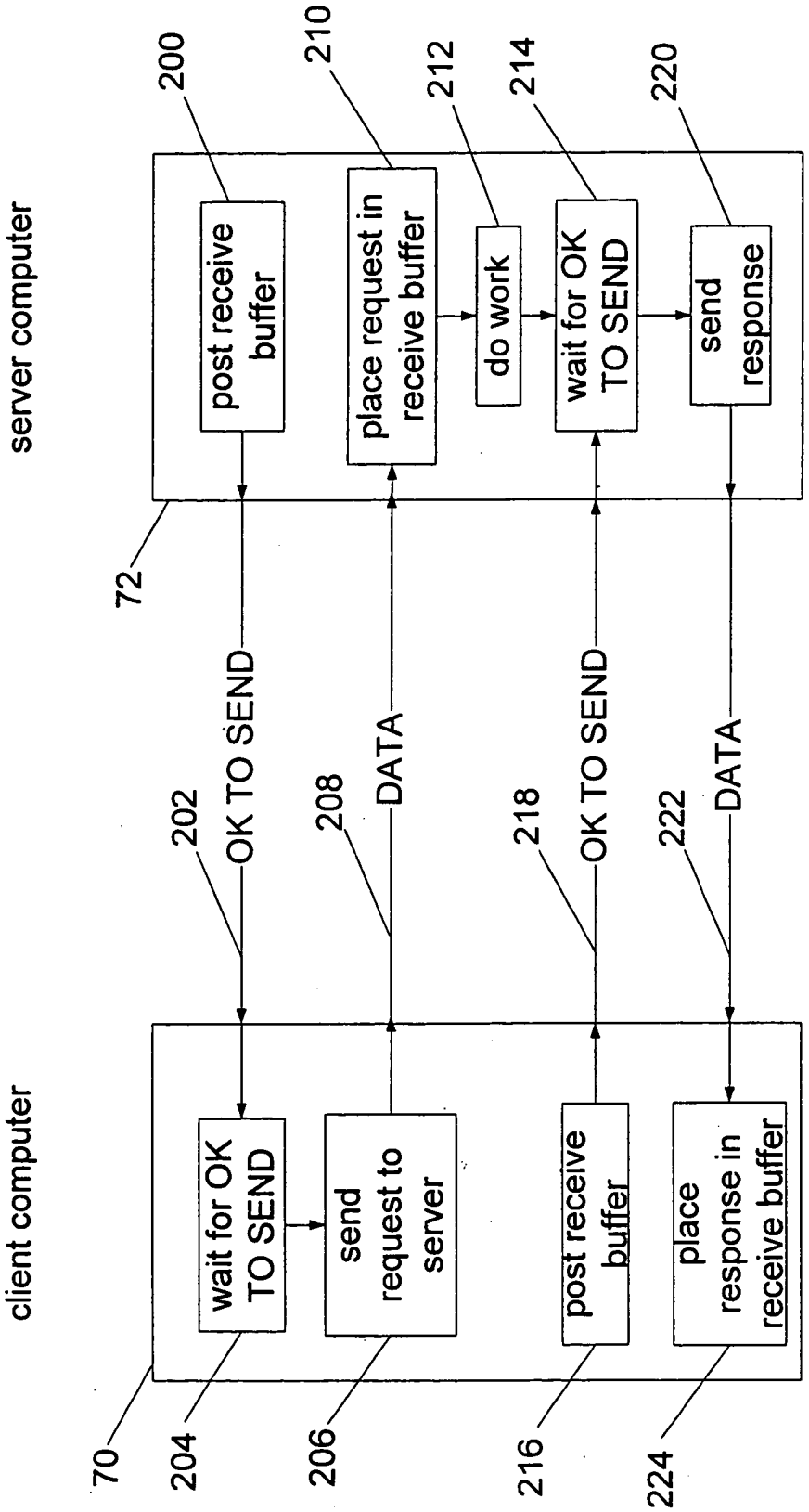
server computer
client computer
post receive buffer
place request in receive buffer
do work
wait for OK TO SEND
send response
wait for OK TO SEND
send request to server
post receive buffer
place response in receive buffer
OK TO SEND
DATA
OK TO SEND
DATA
200
210
212
214
220
72
202
208
218
222
70
204
206
216
224

FIG. 7

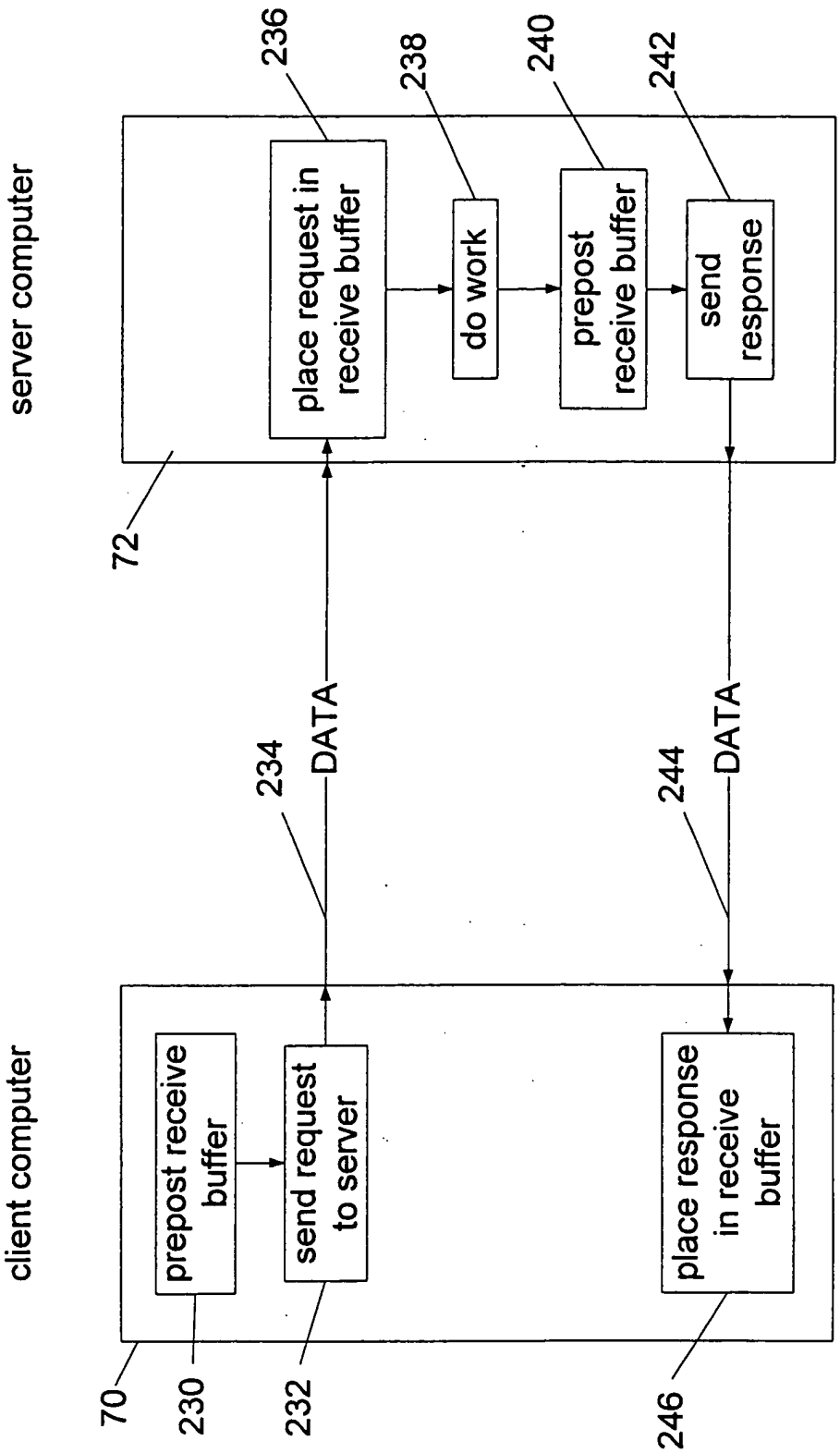
server computer
client computer
place request in receive buffer
do work
prepost receive buffer
send response
prepost receive buffer
send request to server
place response in receive buffer
DATA
DATA
72
70
230
232
234
236
238
240
242
244
246

Fig. 8

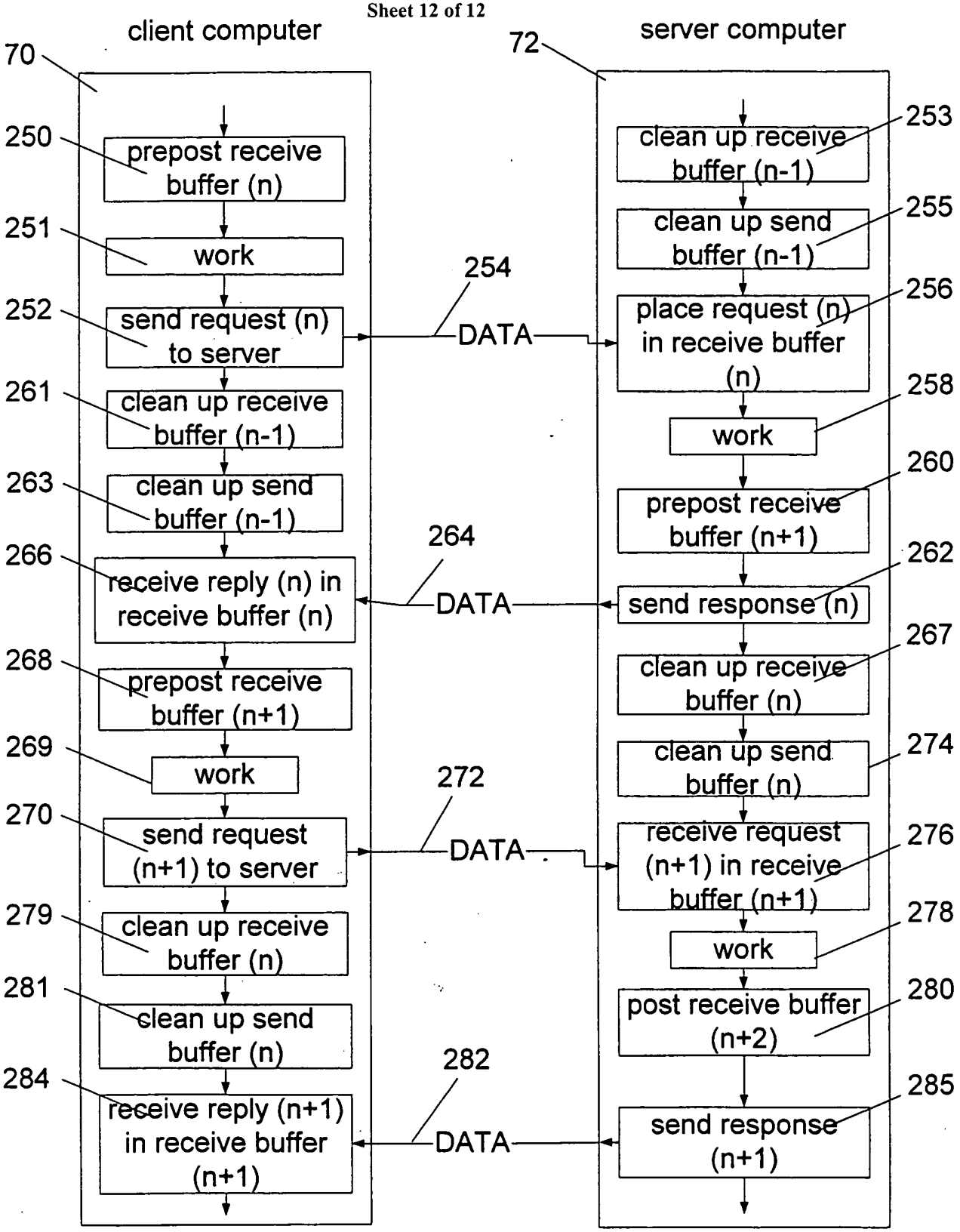
client computer
server computer
70
72
250
prepost receive buffer (n)
251
work
252
send request (n) to server
261
clean up receive buffer (n-1)
263
clean up send buffer (n-1)
266
receive reply (n) in receive buffer (n)
268
prepost receive buffer (n+1)
269
work
270
send request (n+1) to server
279
clean up receive buffer (n)
281
clean up send buffer (n)
284
receive reply (n+1) in receive buffer (n+1)
254
DATA
264
DATA
272
DATA
282
DATA
253
clean up receive buffer (n-1)
255
clean up send buffer (n-1)
256
place request (n) in receive buffer (n)
258
work
260
prepost receive buffer (n+1)
262
send response (n)
267
clean up receive buffer (n)
274
clean up send buffer (n)
276
receive request (n+1) in receive buffer (n+1)
278
work
280
post receive buffer (n+2)
285
send response (n+1)

FIG. 9